

Implementasi PHP sebagai Penghasil Konten Otomatis pada Halaman Situs

Mardi Siswo Utomo

Program Studi Teknik Informatika, Universitas Stikubank
email : mardiutomo@gmail.com

Abstrak

Selain dengan menulisnya secara manual konten situs dapat dihasilkan oleh aplikasi dengan menggunakan penghasil konten. Selain itu bagaimana cara membuat aplikasi penghasil konten pada suatu situs yang dapat bekerja secara otomatis. Masalah dibatasi dengan hanya melakukan implementasi aplikasi pada lingkungan CMS wordpress dan ujicoba pada suatu situs.

Secara garis besar sistem akan mencoba mengambil dokumen dengan mencari kata kunci di situs <http://en.wikipedia.org> jika di temukan halaman akan di scrap, jika tidak cari kata kunci di situs <http://id.wikipedia.org> demikian seterusnya dan terakhir jika tidak ditemukan dokumen dengan kata kunci tersebut akan di lakukan pencarian di mesin pencari <http://searchbing.com> dan <http://yahoo.com>. Teknik Penghasil Konten dapat di implementasikan on the fly sehingga hanya dilakukan saat seorang pengguna membutuhkan dan dokumen yang dimaksud tidak terdapat didalam website. Dari tren pencarian google.com dan google.co.uk pada tanggal 8 februari 2011, dihasilkan 60% halaman dokumen yang berhubungan dengan kata-kata kunci pada trend pencarian google.

Kata kunci : Auto Content Penghasil, SERP, CMS Wordpress

PENDAHULUAN

Keragaman konten, jumlah konten dan keterkaitan diantaranya pada suatu situs merupakan parameter penting untuk menghasilkan nilai SERP yang tinggi. Konten-konten yang unik dan masih dalam suatu pokok permasalahan biasanya dapat terhubung dengan baik (Ast A., 2010).

Butuh usaha yang besar untuk membangun konten yang unik dan bagus, sehingga tidak dapat dilakukan dengan cepat. Penghasil konten merupakan salah satu alternatif untuk meningkatkan SERP situs dengan cepat. Konten-konten dapat dihasilkan secara otomatis dengan bantuan feed atau rss dari mesin pencari

Diharapkan dengan jumlah konten yang banyak, semakin mudah situs untuk tampil di halaman mesin pencari sehingga pengunjung situs dapat ditingkatkan dengan mudah. Walaupun demikian penghasil konten

merupakan cara sementara untuk membuat situs menjadi populer, konten yang bagus dan unik merupakan kunci utama dari keberhasilan suatu situs menarik pengunjung (Hadiono, 2010).

Tulisan ini akan membahas bagaimana membangun aplikasi penghasil konten pada suatu situs yang dapat bekerja secara otomatis. Masalah dibatasi dengan hanya melakukan implementasi aplikasi pada lingkungan CMS wordpress dan ujicoba pada suatu situs.

TUJUAN

Tujuan yang ingin dicapai adalah merancang dan membuat aplikasi penghasil konten berbasis web yang dapat membangun konten pada situs ber-CMS wordpress dengan tujuan meningkatkan SERP situs. Manfaat yang diharapkan adalah : Dapat meningkatkan utilitas dan kemanfaatan suatu situs di internet dengan melihat dari jumlah pengunjung di situs tersebut.

LANDASAN TEORI

Web Konten

Elemen-elemen seperti teks, forms, images (GIFs, JPEGs, Portable Network Graphics) dan video dapat diletakkan didalam halaman menggunakan tag-tag HTML/XHTML/XML. Browser terkadang juga memerlukan Plug-ins seperti Adobe Flash, QuickTime, Java, dan sebagainya untuk menampilkan beberapa media yang diletakkan didalam halaman web menggunakan tag-tag HTML/XHTML.

Halaman web dan situs web dapat berupa halaman statis, atau dapat diprogram secara dinamis sehingga menghasilkan halaman web dengan konten atau tampilan visual yang diinginkan, tergantung pada berbagai faktor, seperti masukan dari pengguna akhir, masukan dari Webmaster, atau perubahan dalam lingkungan komputasi (w3c.com).

XML

XML (Extensible Markup Language) adalah bahasa markup untuk keperluan umum yang disarankan oleh W3C untuk membuat dokumen markup keperluan pertukaran data antar sistem yang beraneka ragam. XML merupakan kelanjutan dari HTML (HyperText Markup Language) yang merupakan bahasa standar untuk melacak Internet.

XML didesain untuk mampu menyimpan data secara ringkas dan mudah diatur. Kata kunci utama XML adalah data (jamak dari datum) yang jika diolah bisa memberikan informasi.

XML menyediakan suatu cara terstandarisasi namun bisa dimodifikasi untuk menggambarkan isi dari dokumen. Dengan sendirinya, XML dapat digunakan untuk menggambarkan sembarang view database, tetapi dengan suatu cara yang standar. XML memiliki tiga tipe file :

- a. XML, merupakan standar format dari struktur berkas (file).
- b. XSL, merupakan standar untuk memodifikasi data yang diimpor atau diekspor.

- c. XSD, merupakan standar yang mendefinisikan struktur database dalam XML.

RSS

RSS adalah sebuah file berformat XML untuk sindikasi yang telah digunakan (diantaranya dan kebanyakan) situs web berita dan weblog. Singkatan ini biasanya mengarah ke beberapa :

- a. Rich Site Summary (RSS 0.91)
- b. RDF Site Summary (RSS 0.9 and 1.0)
- c. Really Simple Syndication (RSS 2.0)

Teknologi yang dibangun dengan RSS memungkinkan kita untuk berlangganan kepada situs web yang menyediakan umpan web (feed) RSS, biasanya situs web yang isinya selalu diganti secara reguler. Untuk memanfaatkan teknologi ini kita membutuhkan layanan pengumpul. Pengumpul bisa dibayangkan sebagai kotak surat pribadi. Kita kemudian dapat mendaftar ke situs yang ingin kita tahu perubahannya. Namun, berbeda dengan langganan koran atau majalah, untuk berlangganan RSS tidak diperlukan biaya, gratis. Tapi, kita biasanya hanya mendapatkan satu baris atau sebuah pengantar dari isi situs berikut alamat terkait untuk membaca isi lengkap artikelnya.

ANALISA DAN PERANCANGAN SISTEM

Spesifikasi Kebutuhan Perangkat Lunak.

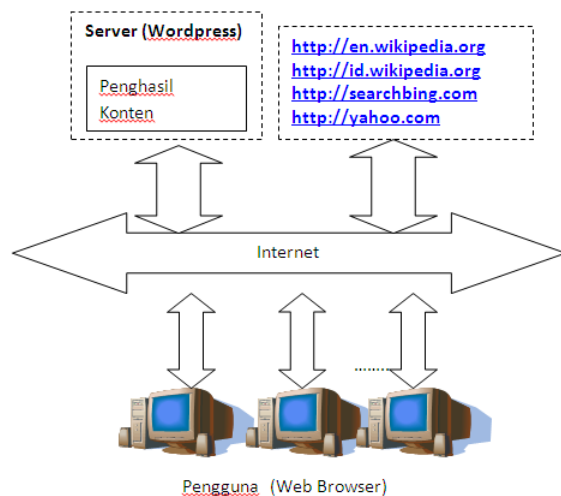
Sistem ini adalah Rekayasa Perangkat Lunak Komputer berbasis web yang bertujuan untuk menghasilkan konten secara otomatis dari suatu kata kunci. Hal-hal yang diharapkan oleh pengguna agar dapat diwujudkan dalam sistem ini diantaranya adalah hal-hal sebagai berikut :

- a. Sistem dapat secara otomatis menghasilkan suatu konten yang nantinya akan ditampilkan di halaman web, berdasarkan pada suatu kata kunci.
- b. Untuk mempersingkat penelitian sehingga fokus pada penelitian penghasil konten maka digunakan CMS wordpress untuk menangani proses manajemen website.

- c. Sistem dapat secara otomatis menampilkan konten hasil penghasil dalam bentuk halaman web site atau memasukan hasil tersebut ke dalam artikel di wordpress, baik untuk judul, isi, rangkuman maupun kata kunci / tag.
- d. Sistem ini dapat diintegrasikan dengan plugin wordpress yang lain sehingga mempermudah instalasi.

Dalam pengembangan aplikasi ini diharapkan dapat memberikan manfaat sbb :

- a. Sistem dapat menggenerasi konten utama dengan bantuan mesin pencari.
- b. Sistem dapat membuat konten secara otomatis sesuai kebutuhan pada website dengan basis wordpress.



Gambar 1 Perspektif produk

Perspektif produk

Aplikasi yang dibangun ditanam pada web server yang terkoneksi dengan jaringan internet. Aplikasi berjalan menggunakan service http dengan format transaksi data html, sehingga dapat dibuka menggunakan terminal yang terkoneksi ke jaringan komputer dan mampu / mempunyai browser WEB. Pengguna dapat melihat konten yang telah dihasilkan dalam bentuk artikel dalam wordpress.

Pada gambar 1 diperlihatkan Komputer Server berfungsi sebagai web server yang terinstall wordpress. Web server akan mencoba mengambil dokumen dengan mencari kata kunci

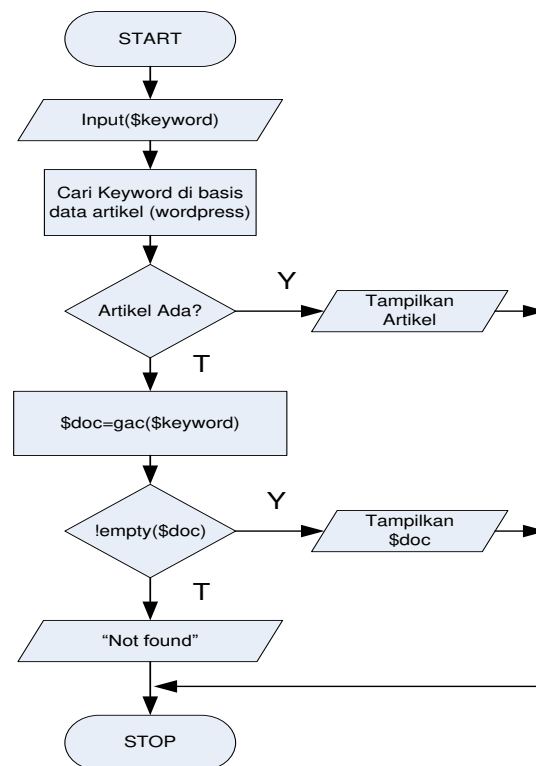
di situs <http://en.wikipedia.org> jika di temukan halaman akan di scrap, jika tidak cari kata kunci di situs <http://id.wikipedia.org> demikian seterusnya dan terakhir jika tidak ditemukan dokumen dengan kata kunci tersebut akan di lakukan pencarian di mesin pencari <http://searchbing.com> dan <http://yahoo.com>.

Fungsi-fungsi Produk

Produk Aplikasi dibangun dengan antarmuka web, sehingga semua fungsi dapat langsung diakses dari halaman aktif manapun. Walaupun demikian tidak ada menu atau fungsi yang secara eksplisit merujuk ke sistem penghasil konten

Diagram Alir Pencarian Artikel pada Wordpress

Aliran Proses pencarian artikel pada website berbasis wordpress dengan Penghasil Konten terinstall diperlihatkan pada gambar 2. Variabel keyword dimasukan sebagai dasar pencarian artikel, jika artikel di temukan di dalam basis data maka artikel akan di tampilkan.

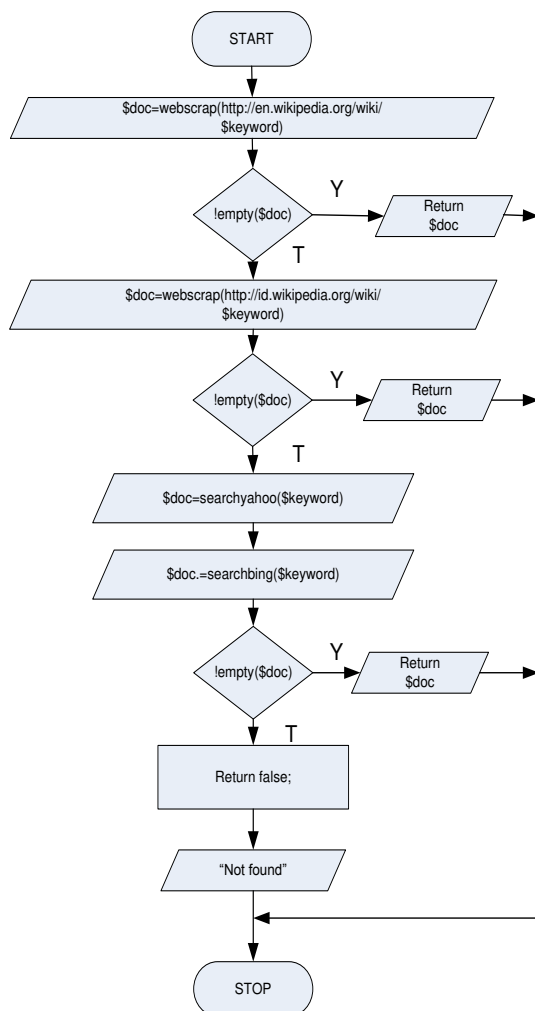


Gambar 2. Diagram aliran proses menu utama aplikasi

Jika artikel tidak ditemukan maka dilakukan proses generasi konten untuk kata kunci yang dimasukan. Jika penghasil konten berhasil menemukan artikel maka akan ditampilkan ke halaman web apabila tidak maka akan ditampilkan error Not Found.

Diagram Alir Fungsi Penghasil Konten

Aliran proses penghasil konten ditampilkan pada gambar 3. Fungsi akan memulai dengan mengambil halaman web pada url [http://en.wikipedia.org/wiki/\\$keyword](http://en.wikipedia.org/wiki/$keyword). Kemudian dilakukan proses scraping pada halaman tersebut jika di temukan sistem berhasil menghasilkan dokumen yang berhubungan dengan keyword dan mengembalikannya dalam bentuk hasil fungsi.



Gambar 3. Diagram Aliran Proses Fungsi Penghasil Konten

Jika gagal menemukan dokumen dari [http://en.wikipedia.org/wiki/\\$keyword](http://en.wikipedia.org/wiki/$keyword) maka dilakukan pengambilan dokumen di situs selanjutnya yaitu [http://id.wikipedia.org/wiki/\\$keyword](http://id.wikipedia.org/wiki/$keyword).

Kemudian dilakukan proses scraping pada halaman tersebut jika di temukan sistem berhasil menghasilkan dokumen yang berhubungan dengan keyword dan mengembalikannya dalam bentuk hasil fungsi.

Jika gagal menemukan dokumen dari [http://id.wikipedia.org/wiki/\\$keyword](http://id.wikipedia.org/wiki/$keyword) maka dilakukan pengambilan dokumen dari mesin pencari dengan hasil berupa dokumen berformat xml / rss.

HASIL DAN PEMBAHASAN

Wikipedia Scraping

Pada gambar 3 diperlihatkan alur proses dari fungsi Penghasil Konten, dimana fungsi terlebih dahulu melakukan pengambilan halaman web di website en.wikipedia.org. Proses pengambilan halaman atau biasa disebut dengan fetching dapat dilakukan dengan perintah PHP `file_get_content`, tetapi pada kasus en.wikipedia.org metode `file_get_content` tidak dapat dilakukan dikarenakan adanya leech detection di webserver wikipedia.

Metode lainnya untuk mengambil konten web adalah CURL, metode ini dapat mengirim informasi lengkap dan detil layaknya sebuah web browser sehingga web server menganggap permintaan dilakukan oleh seorang pengguna dengan menggunakan web browser. Pada penelitian ini tidak secara langsung menggunakan perintah CURL untuk mengambil halaman web, tetapi menggunakan pustaka snoop. Pustaka snoop berisi fungsi-fungsi PHP untuk melakukan fetch di suatu halaman web dengan perintah dasar CURL.

Pada program 1 diperlihatkan potongan kode php yang melakukan fetch, sebelum melakukan fetch terlebih dahulu dilakukan deklarasi obyek `$snoop` dari kelas Snoop, kemudian user agent string di set opera sehingga web server akan mengenali fungsi sebagai browser opera. Sebelum keyword di gunakan terlebih dahulu di filter menjadi huruf kecil

semua dan mengganti karakter + menjadi karakter spasi.

Program 1. Implementasi Operasi Fetch Halaman WEB en.wikipedia.org

```
$snoopy = new Snoopy;
$snoopy->agent='opera';
$keyword=strtolower($keyword);
$keyw=str_replace(" ", " ",
", $keyword);
$keyw=str_replace("
", "+", $keyword);
$uristr="http://en.wikipedia.org/w/index.php?title=Special%3ASearch&search=$keyword";
$snoopy->fetch($uristr);
$striklan= $snoopy->results;
```

Pada program 2 . Konten utama dibatasi dari header oleh teks "<!-- bodytext -->" dan dibatasi oleh footer oleh "<!-- /bodytext -->", sehingga pada program 3 digunakan perintah regular ekspresi dengan mendeteksi pola <!-- bodytext -->(.*)<!-- /bodytext -->.

Program 2. Pola Konten utama halaman dokumen wikipedia

```
...bagian header
<!-- bodytext -->
.
.
Konten Utama
..
.
<!-- /bodytext -->
....bagian footer
```

Pada program 3 diperlihatkan operasi Penghasil Konten dengan menggunakan metode regular ekspresi. Konten utama halaman web hasil en.wikipedia.com diapit oleh bagian header dan footer seperti terlihat

Program 3. Perintah ekspresi regular untuk memisahkan konten utama wikipedia

```
$str= $snoopy->results;
ereg("<!-- bodytext -->(.*)<!-- /bodytext -->", $str, $match);
$str=$match[0];
```

Search Engine Scraping

Pada program 4 diperlihatkan potongan kode php yang melakukan fetch kata kunci dari API yahoo search. Yahoo menyediakan API

untuk hasil pencariannya pada alamat url <http://api.search.yahoo.com/WebSearchService/rss/webSearch.xml>, hasil dari API Yahoo search adalah dokumen berupa XML dengan standar rss feed.

Pada program 5 diperlihatkan potongan kode php yang melakukan fetch kata kunci dari API bing search. Bing menyediakan API untuk hasil pencariannya pada alamat <http://www.bing.com/search?go=&form=QBLH&filt=all&format=rss>, hasil dari API bing search adalah dokumen berupa XML dengan standar rss feed.

Program 4 program untuk mengambil feed dari yahoo search API

```
$snoopy = new Snoopy;
$snoopy->agent='opera';
$keyword=strtolower($keyword);
$keyw=str_replace(" ", " ",
", $keyword);
$keyw=str_replace("
", "+", $keyword);
$uristr="http://api.search.yahoo.com/WebSearchService/rss/webSearch.xml?appid=yahoosearchwebrss&query=$keyword";
$snoopy->fetch($uristr);
$str= $snoopy->results;
```

Program 5 program untuk mengambil feed dari bing search API

```
$snoopy = new Snoopy;
$snoopy->agent='opera';
$keyword=strtolower($keyword);
$keyw=str_replace(" ", " ",
", $keyword);
$keyw=str_replace("
", "+", $keyword);
$uristr="http://www.bing.com/search?q=$keyword&go=&form=QBLH&filt=all&format=rss";
$snoopy->fetch($uristr);
$str= $snoopy->results;
```

IMPLEMENTASI

Implementasi Wordpress Plugin

Setelah fungsi selesai ditulis maka untuk mempermudah penggunaan dan integrasi dengan wordpress maka struktur program fungsi Penghasil Konten di rubah menjadi struktur plugin pada wordpress.

Struktur program plugin pada wordpress mengharuskan ditambahkan header remark yang berfungsi untuk memuat informasi seputar plugin tersebut, seperti terlihat pada potongan program 6.

Program 6 Header plugin wordpress Penghasil Konten

```

Plugin Name: Penghasil Konten
Plugin URI:
http://unisbank.ac.id
Description: menghasilkan
konten secara otomatis
Version: 1.0
Author: FTI
Author URI:
http://unisbank.ac.id

```

Instalasi Plugin Penghasil Konten

Plugin pada wordpress diinstall melalui menu administrator di url [http://\[namadomain\]/wp-admin/](http://[namadomain]/wp-admin/) setelah terlebih dahulu memasukan username dan password untuk administrator. Plugin dipasang pada menu `plugins->add new`, plugin dapat di upload ataupun langsung di download dari repository wordpress.

Pada penelitian ini plugin tidak terdapat di repository wordpress, sehingga digunakan menu upload untuk menambahkan plugin. Plugin akan aktif setelah diaktifkan dengan meng klik url `activate` dibawah nama plugin. Pada gambar 4.2 diperlihatkan tampilan layar daftar plugin yang terpasang di wordpress.



Gambar 3 Tampilan layar daftar plugin wordpress

Aktifasi Fungsi Penghasil Konten

Setelah plugin terpasang tidak serta merta fungsi Penghasil Konten telah digunakan pada wordpress. Untuk dapat berfungsi seperti proses yang digambarkan pada gambar 4 maka dibutuhkan modifikasi pada bagian template / theme yang digunakan. Pada gambar 4 diperlihatkan bahwa proses yang di sisipi fungsi Penghasil Konten adalah proses pencarian.

Pada wordpress proses pencarian melibatkan file `search.php` pada theme yang aktif untuk menghasilkan halaman konten. File `search.php` pada bagian tampilan "Not Found" di ganti dengan pemanggilan fungsi Penghasil Konten sehingga sebelum menampilkan not found terlebih dahulu dilakukan pencarian pada situs en.wikipedia.org, seperti terlihat pada program 7

Program 7. Perubahan pada search.php

```

<?php if ( have_posts() ) { ?>
<h1 class="page-title"><?php
printf( __( 'Search Results for:
%s', 'twentyten' ), '<span>' .
get_search_query() . '</span>' );
?></h1>
<?php } else {
global $s;
include "wp-includes/class-
snoopy.php";
if((function_exists('penghasilk
onten')) and (!empty($s))) $doc=
penghasilkonten ($s);
if(strlen($doc)>500) echo "<div
id='post-0' class='post no-results
not-found'>
<h2
class='entry-title'>$s</h2>
<div
class='entry-content'>
<p>$doc</p>
</div><!-- .entry-content -->
</div><!--
#post-0 -->";
else { ?>
<div id="post-0" class="post
no-results not-found">
<h2 class="entry-title"><?php
_e( 'Nothing Found', 'twentyten' );
?></h2>
<div
class="entry-content">
<p><?php _e( 'Sorry, but
nothing matched your search
criteria. Please try again with
some different keywords.',
'twentyten' ); ?></p>
<?php get_search_form(); ?>
</div><!-- .entry-content -->
</div><!--
#post-0 -->
<?php } } ?>

```

KESIMPULAN

Teknik Penghasil Konten dapat di implementasikan on the fly sehingga hanya dilakukan saat seorang pengguna membutuhkan dan dokumen yang dimaksud tidak terdapat didalam website. Dari tren pencarian google.com dan google.co.uk pada tanggal 8 februari 2011, dihasilkan 60% halaman dokumen yang berhubungan dengan kata-kata kunci pada trend pencarian google.

Untuk dapat di rangking dengan bagus oleh mesin pencari dibutuhkan konten unik, karena dengan teknik Penghasil Konten, konten yang dihasilkan tidak unik maka teknik ini website tidak dapat menjadi urutan awal pada hasil mensin pencari.

Untuk menghasilkan artikel yang unik perlu digabungkan dengan teknik text sumarization ataupun random sentences producer dan perbaikan di nilai meta.

Masalah hukum dan HAKI menjadi masalah utama pada situs yang menggunakan teknik Penghasil Konten ini. Tetapi dapat di tambahkan pada footer sumber asli dari dokumen tersebut.

DAFTAR PUSTAKA

- Ast A., (2010), *The fundamentals of SEO for average joe*
- Hadiono, K., (2010), *Aplikasi Web Scrapping Untuk Koleksi Konten Utama [1] Halaman Situs*, Unisbank
- JISC Briefing Paper, 2006, *Text mining*, JISC, Inggris
- Murhadin, E., (2003), *PHP Programming Fundamental dan MySQL Fundamental*, <http://ikc.cbn.net.id/umum/andy-php.php>
- Nugroho, Bunafit, (2004), *PHP & MySQL Dengan Editor Dreamweaver MX*, Andi, Yogyakarta
- Pressman R, (1997), *Software Engineering*, Mc Graw Hill, USA
- Prothelon's, (2005), *Web Desain, PHP Programming, Language Learning*, <http://prothelon.com/mambo/tutorial>

W3C, (n.d.), In W3C, diakses tanggal 8 maret 2011, dari <http://www.w3c.org>,